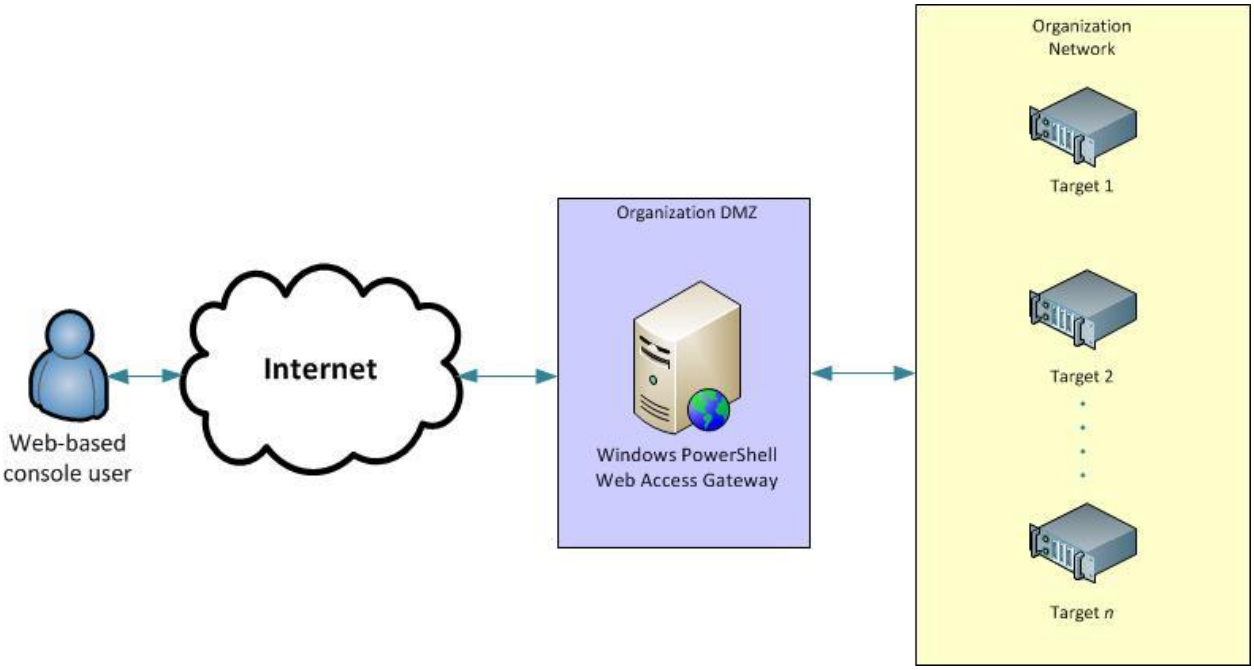
Windows PowerShell Web Access

1. Introduction



From Windows Server 2012. It enables IT Pros to run Windows PowerShell commands and scripts from a Windows PowerShell console in a web browser, with no Windows PowerShell, remote management software, or browser plug-in installation necessary on the client device. All that is required to run the web-based Windows PowerShell console is a properly configured Windows PowerShell Web Access gateway, and a client device browser that supports JavaScript and accepts cookies.

Examples of client devices include laptops, non-work personal computers, borrowed computers, tablet computers, web kiosks, computers that are not running a Windows-based operating system, and cell phone browsers. IT Pros can perform critical

management tasks on remote Windows-based servers from devices that have access to an Internet connection and a web browser.

After successful gateway setup and configuration, users can access a Windows PowerShell console by using a web browser. When users open the secured Windows PowerShell Web Access website, they can run a web-based Windows PowerShell console after successful authentication.

2. Requirements for running Windows PowerShell Web Access

Windows PowerShell Web Access requires Web Server (IIS), .NET Framework 4.5, and Windows PowerShell 3.0 or Windows PowerShell 4.0 to be running on the server on which you want to run the gateway. You can install Windows PowerShell Web Access on a server that is running Windows Server 2012 R2 or Windows Server 2012 by using either the Add Roles and Features Wizard in Server Manager, or Windows PowerShell deployment cmdlets for Server Manager. When you install Windows PowerShell Web Access by using Server Manager or its deployment cmdlets, required roles and features are automatically added as part of the installation process.

Windows PowerShell Web Access allows remote users to access computers in your organization by using Windows PowerShell in a web browser. Although Windows PowerShell Web Access is a convenient and powerful management tool, the web-based

access poses security risks, and should be configured as securely as possible. We recommend that administrators who configure the Windows PowerShell Web Access gateway use available security layers, both the cmdlet-based authorization rules included with Windows PowerShell Web Access, and security layers that are available in Web Server (IIS) and third-party applications. This documentation includes both unsecure examples that are only recommended for test environments, as well as examples that are recommended for secure deployments.

2.1. Browser and client device support

2.1.1. Supported desktop computer browsers

- Windows Internet Explorer for Microsoft Windows 8.0, 9.0, 10.0, and 11.0
- Mozilla Firefox 10.0.2
- Google Chrome 17.0.963.56m for Windows
- Apple Safari 5.1.2 for Windows
- Apple Safari 5.1.2 for Mac OS

2.1.2. Minimally-tested mobile devices or browsers

- Windows Phone 7 and 7.5
- Google Android WebKit 3.1 Browser Android 2.2.1 (Kernel 2.6)
- Apple Safari for iPhone operating system 5.0.1
- Apple Safari for iPad 2 operating system 5.0.1

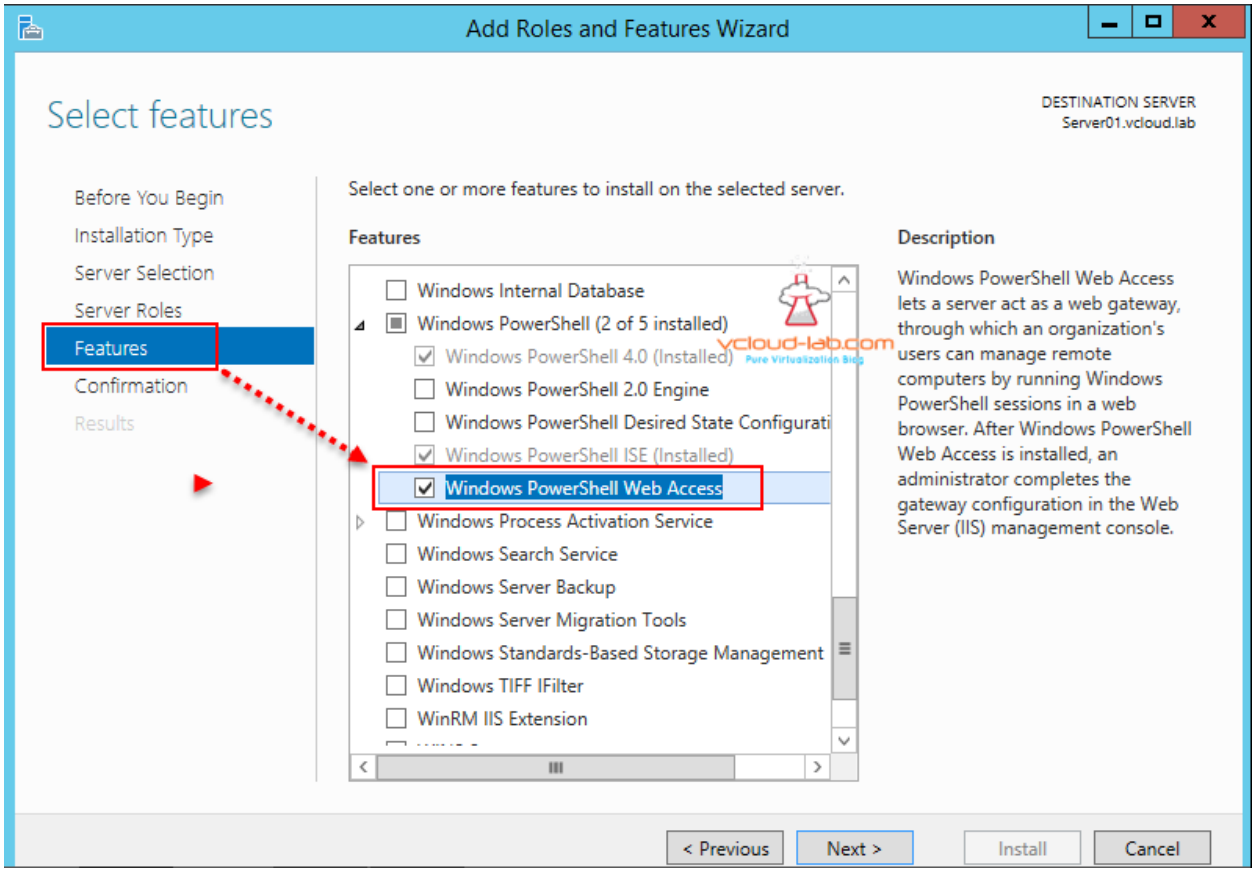
2.1.3. Browser requirements

To use the Windows PowerShell Web Access web-based console, browsers must do the following.

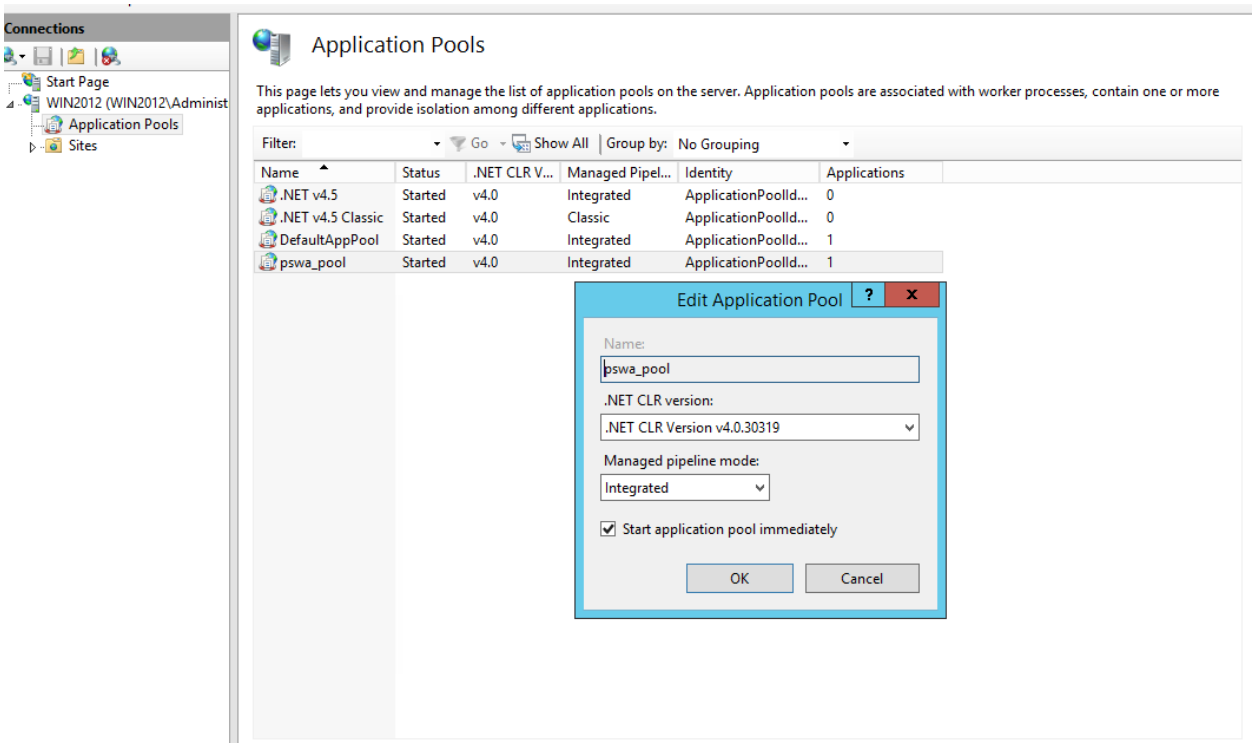
- Allow cookies from the Windows PowerShell Web Access gateway website.
- Be able to open and read HTTPS pages.
- Open and run websites that use JavaScript.

3. Implementation steps

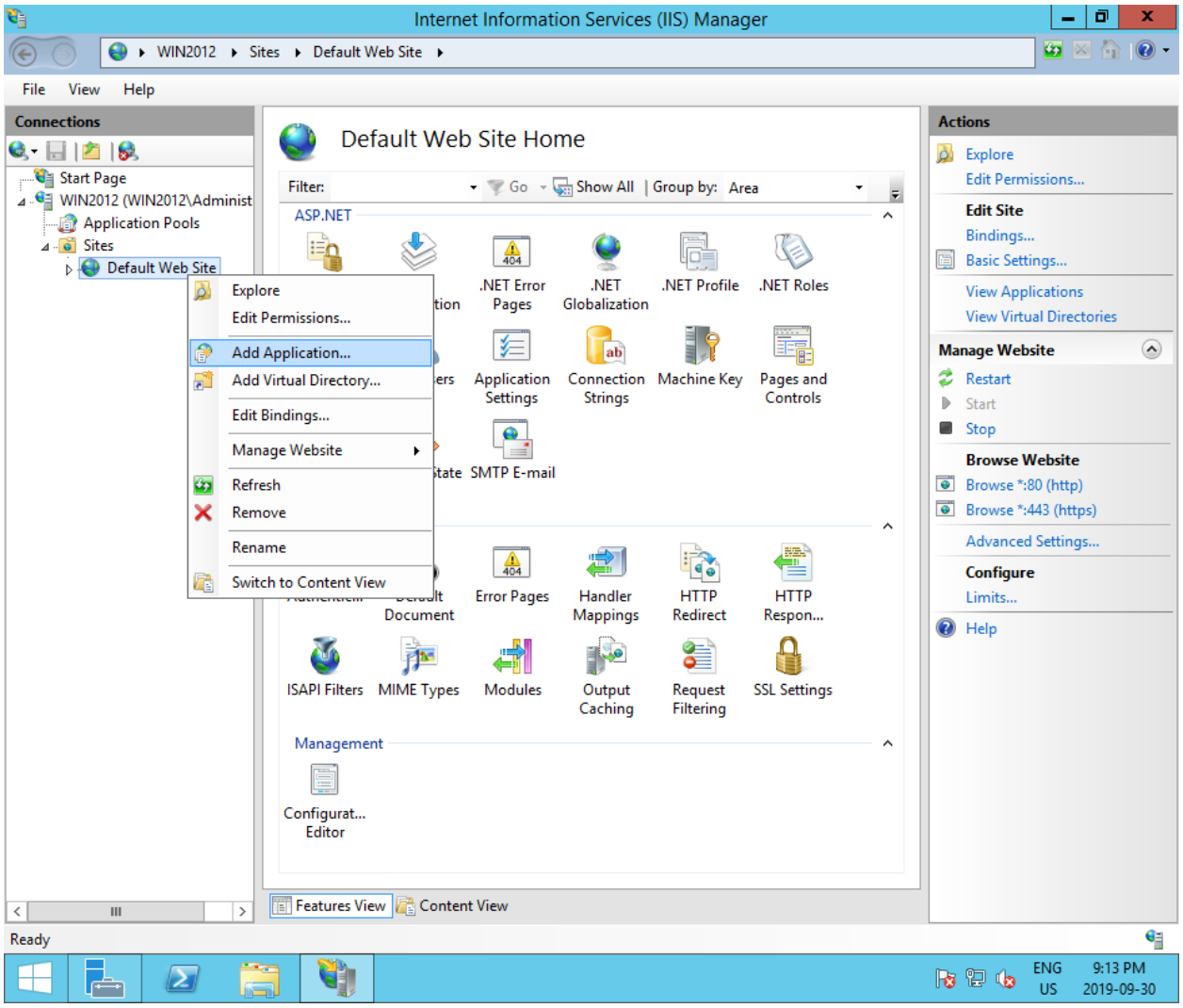
3.1. Add “Windows PowerShell Web Access” from GUI

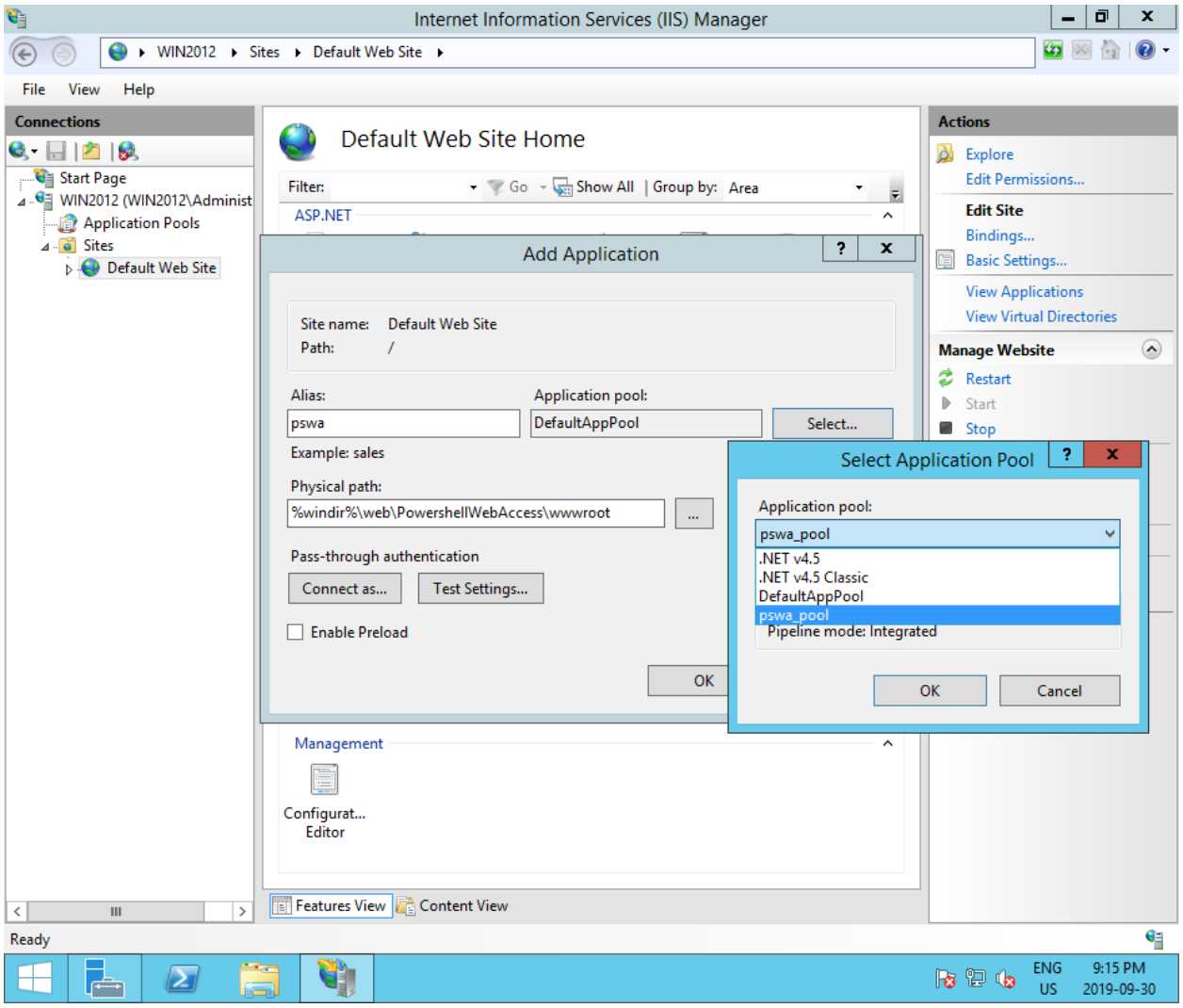


3.2 Add a new Application Pool named “pswa_pool”

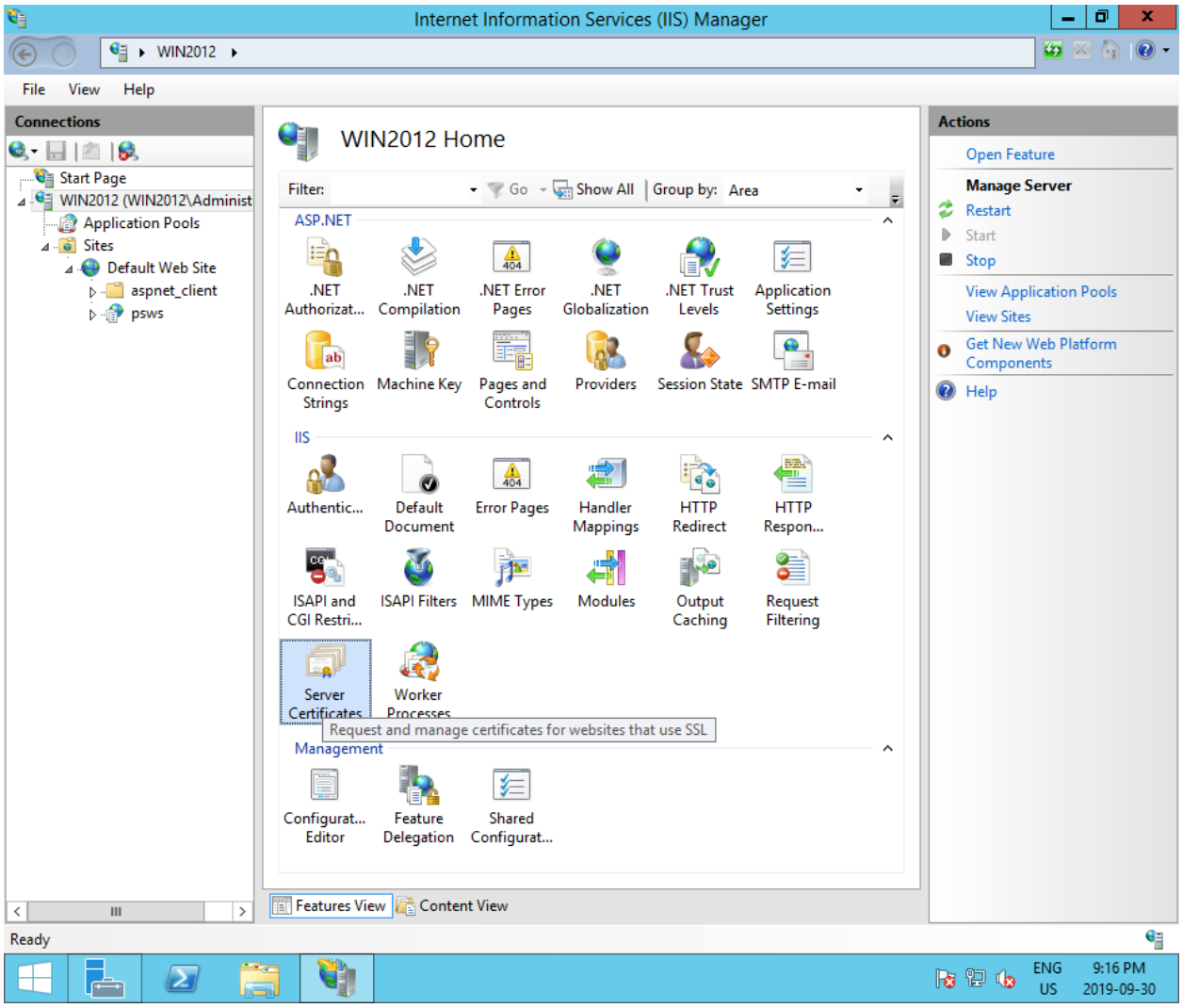


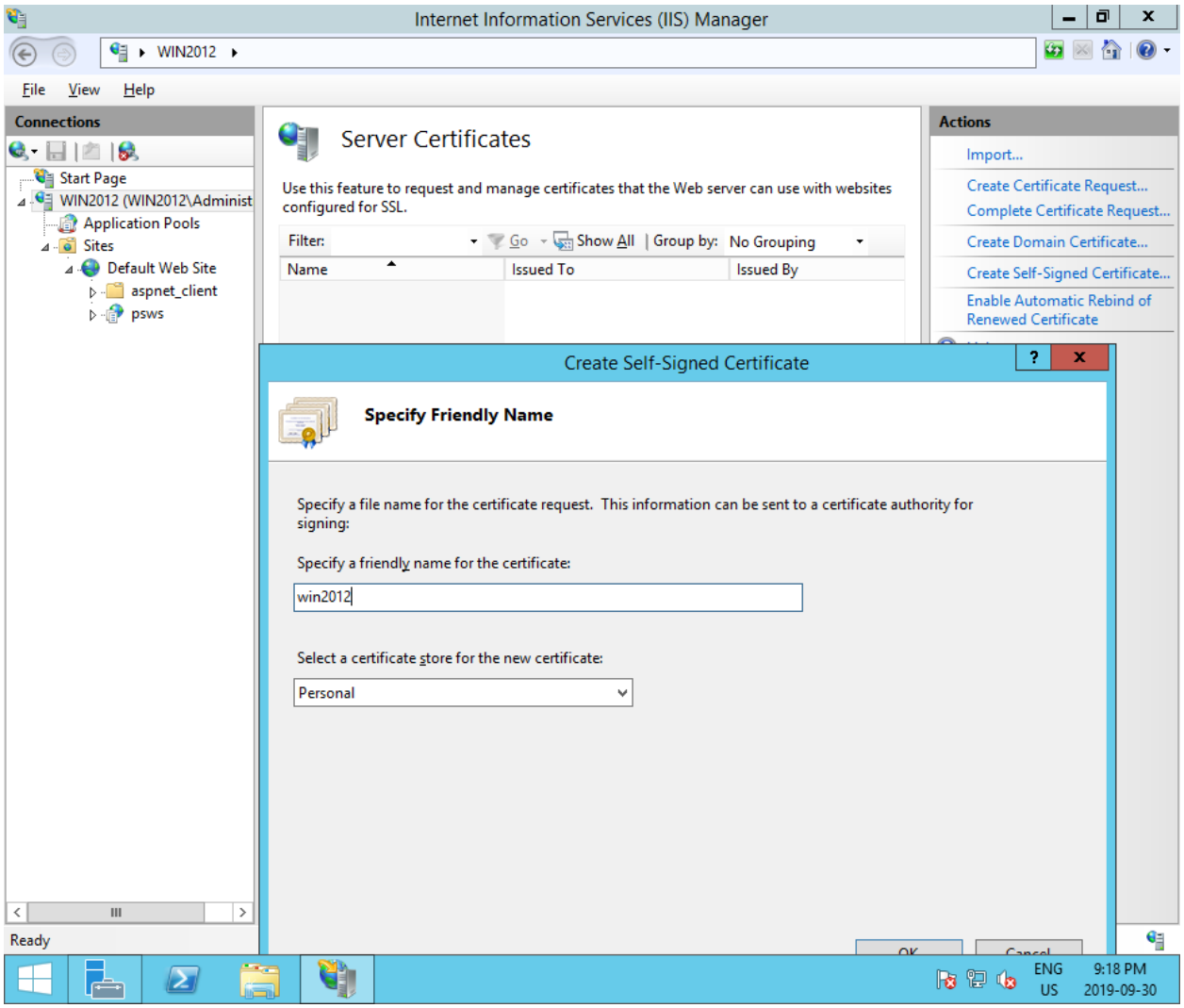
3.3 Apply “pswa_pool” to default web site

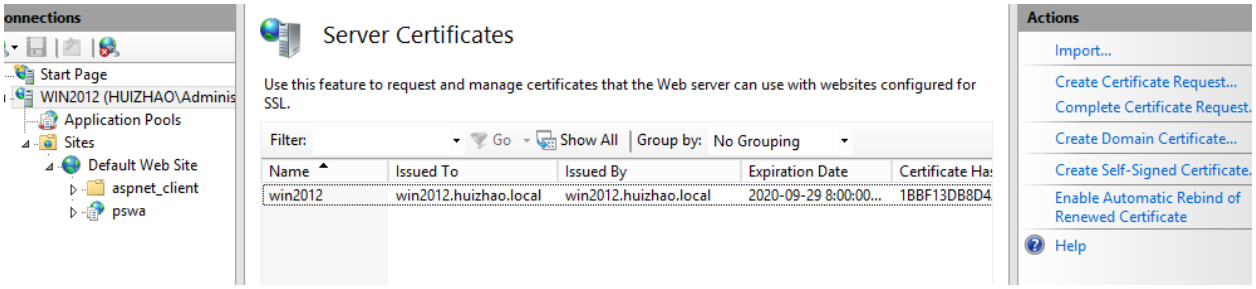

Internet Information Services (IIS) Manager
WIN2012 ▸ Sites ▸ Default Web Site ▸
File View Help
Connections
Start Page
WIN2012 (WIN2012\Administ
Application Pools
Sites
Default Web Site
Explore
Edit Permissions...
Add Application...
Add Virtual Directory...
Edit Bindings...
Manage Website
Refresh
Remove
Rename
Switch to Content View
Default Web Site Home
Filter:
Go
Show All
Group by: Area
ASP.NET
.NET Error Pages
.NET Globalization
.NET Profile
.NET Roles
Application Settings
Connection Strings
Machine Key
Pages and Controls
SMTP E-mail
Error Pages
Handler Mappings
HTTP Redirect
HTTP Respon...
ISAPI Filters
MIME Types
Modules
Output Caching
Request Filtering
SSL Settings
Management
Configurat... Editor
Features View
Content View
Actions
Explore
Edit Permissions...
Edit Site
Bindings...
Basic Settings...
View Applications
View Virtual Directories
Manage Website
Restart
Start
Stop
Browse Website
Browse *:80 (http)
Browse *:443 (https)
Advanced Settings...
Configure
Limits...
Help
Ready
ENG
US
9:13 PM
2019-09-30



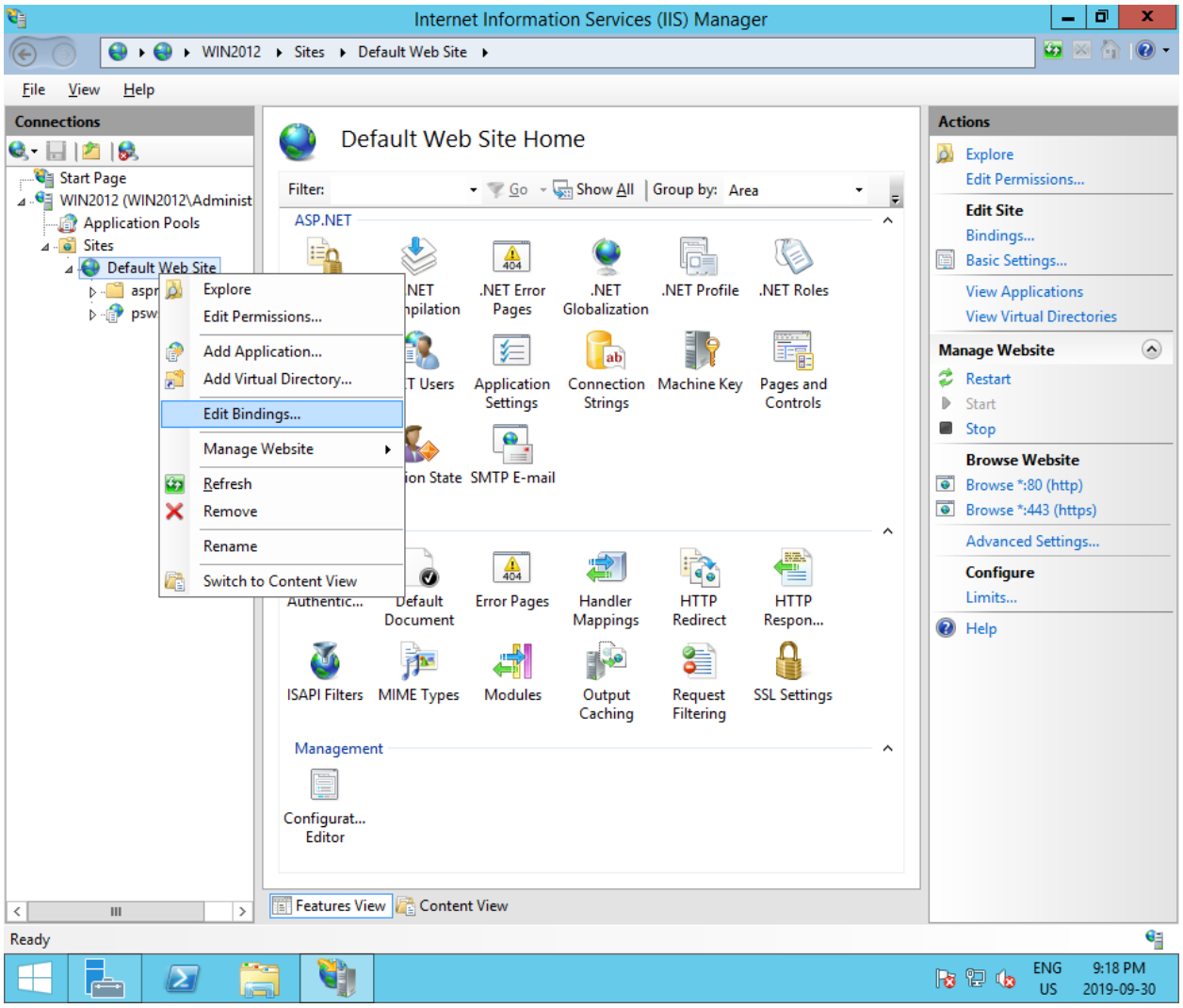
3.4 Add “Server Self-Sign Certification”

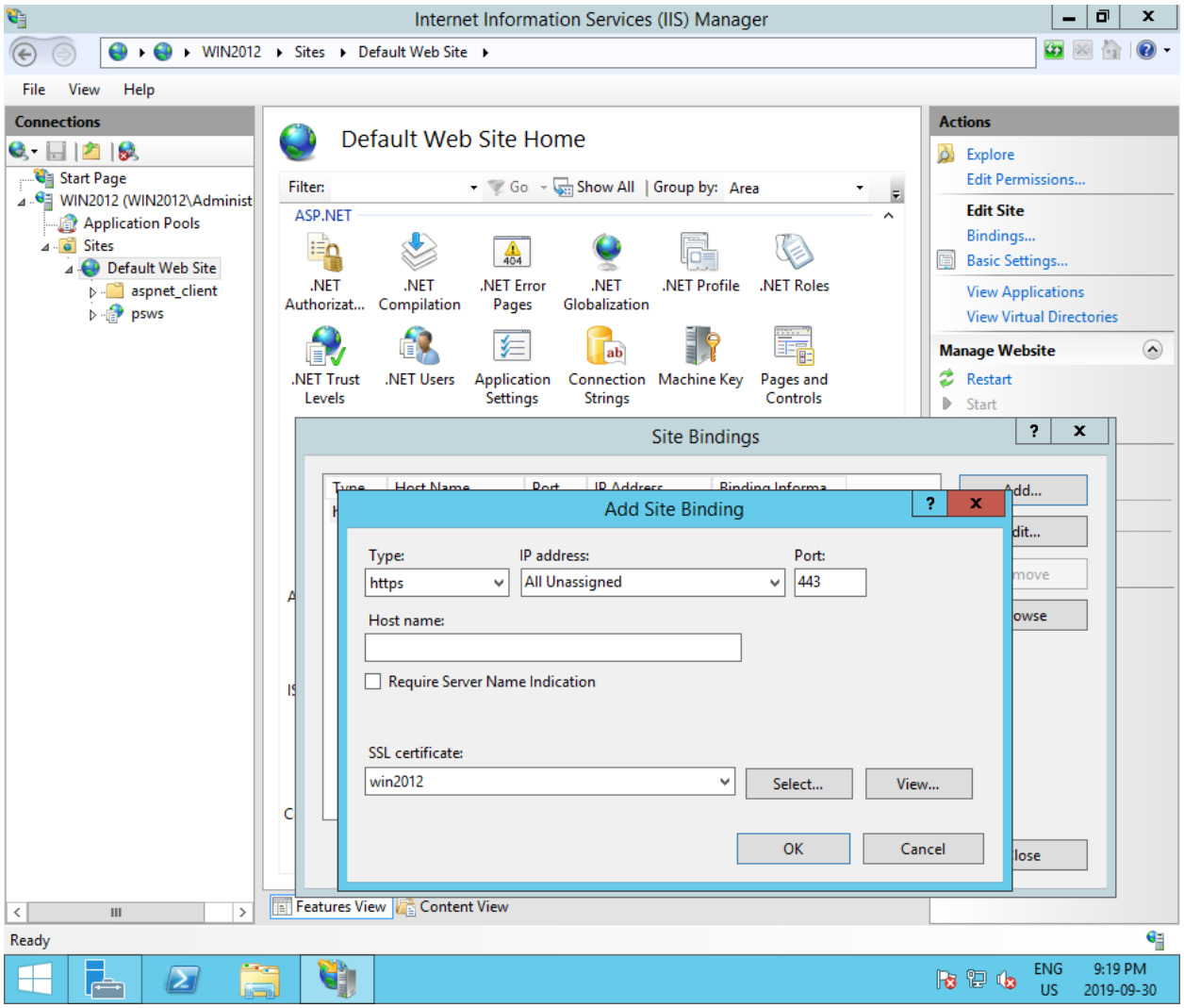

Internet Information Services (IIS) Manager
WIN2012
File View Help
Connections
Start Page
WIN2012 (WIN2012\Administ
Application Pools
Sites
Default Web Site
aspnet_client
psws
WIN2012 Home
Filter:
Go
Show All
Group by: Area
ASP.NET
.NET Authorizat...
.NET Compilation
.NET Error Pages
.NET Globalization
.NET Trust Levels
Application Settings
Connection Strings
Machine Key
Pages and Controls
Providers
Session State
SMTP E-mail
IIS
Authentic...
Default Document
Error Pages
Handler Mappings
HTTP Redirect
HTTP Respon...
ISAPI and CGI Restri...
ISAPI Filters
MIME Types
Modules
Output Caching
Request Filtering
Server Certificates
Worker Processes
Request and manage certificates for websites that use SSL
Management
Configurat... Editor
Feature Delegation
Shared Configurat...
Features View
Content View
Actions
Open Feature
Manage Server
Restart
Start
Stop
View Application Pools
View Sites
Get New Web Platform Components
Help
Ready
ENG US
9:16 PM 2019-09-30





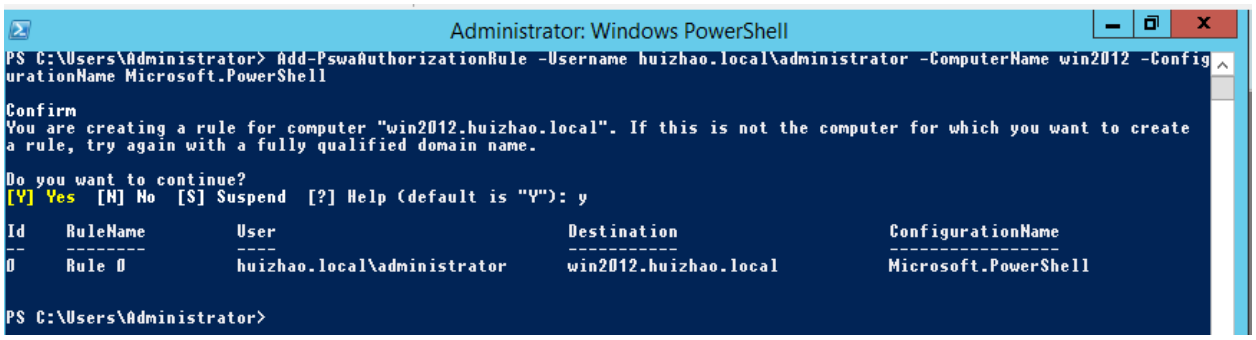
3.5 Bind “SSL credential” to website


Internet Information Services (IIS) Manager
WIN2012 ▸ Sites ▸ Default Web Site ▸
File View Help
Connections
Start Page
WIN2012 (WIN2012\Administ
Application Pools
Sites
Default Web Site
Explore
Edit Permissions...
Add Application...
Add Virtual Directory...
Edit Bindings...
Manage Website
Refresh
Remove
Rename
Switch to Content View
Default Web Site Home
Filter:
Go
Show All
Group by: Area
ASP.NET
.NET Error Pages
.NET Globalization
.NET Profile
.NET Roles
Application Settings
Connection Strings
Machine Key
Pages and Controls
SMTP E-mail
Default Document
Error Pages
Handler Mappings
HTTP Redirect
HTTP Respon...
ISAPI Filters
MIME Types
Modules
Output Caching
Request Filtering
SSL Settings
Management
Configurat... Editor
Features View
Content View
Actions
Explore
Edit Permissions...
Edit Site
Bindings...
Basic Settings...
View Applications
View Virtual Directories
Manage Website
Restart
Start
Stop
Browse Website
Browse *:80 (http)
Browse *:443 (https)
Advanced Settings...
Configure
Limits...
Help
Ready
ENG
US
9:18 PM
2019-09-30



3.6 Give user privilege to remote access windows powershell by using command:

Add-PswaAuthorizationRule -Username huizhao.local\administrator -ComputerName win2012 -ConfigurationName Microsoft.PowerShell



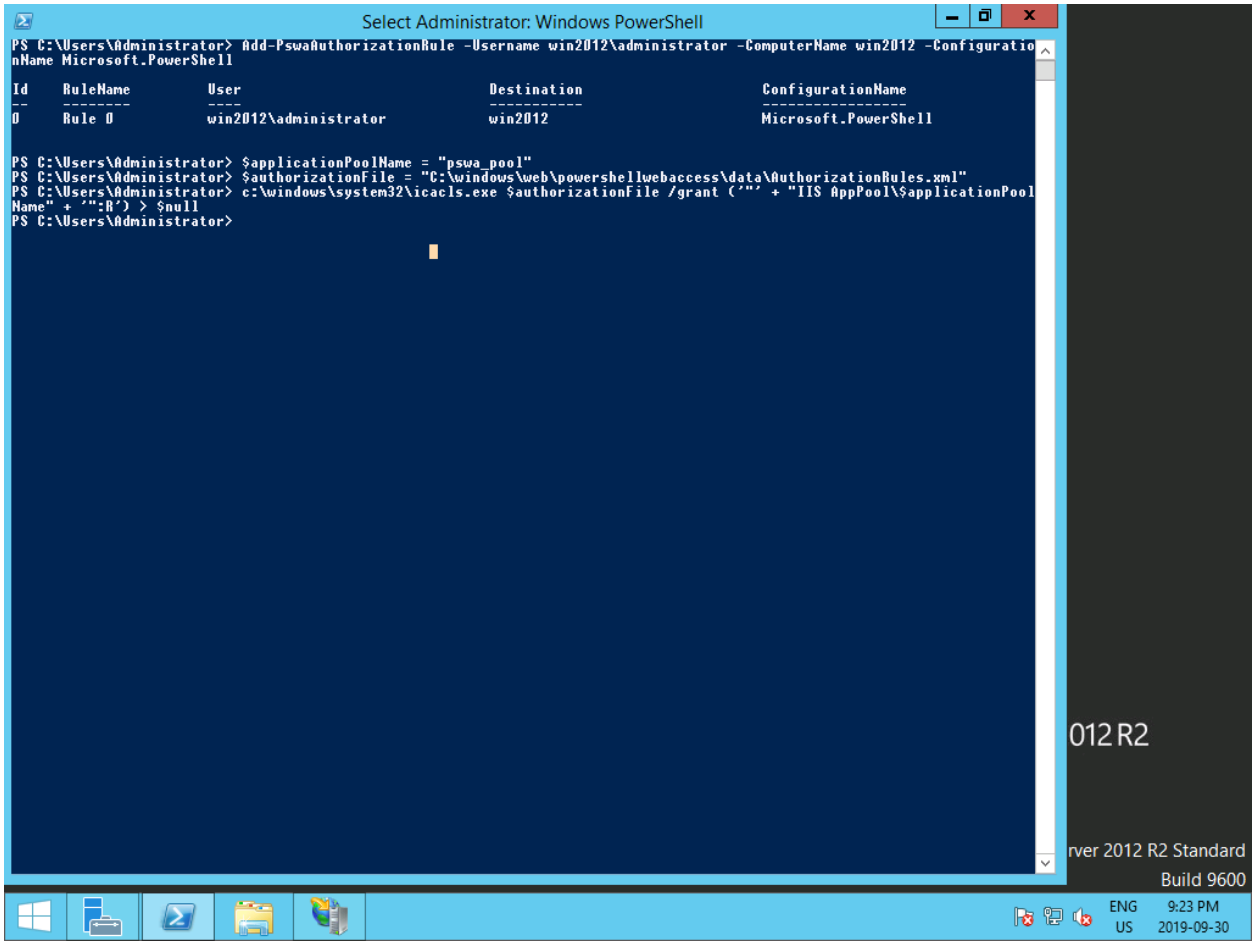
3.7 give web-app privilege to access the authorization config file

$applicationPoolName = "pswa_pool"

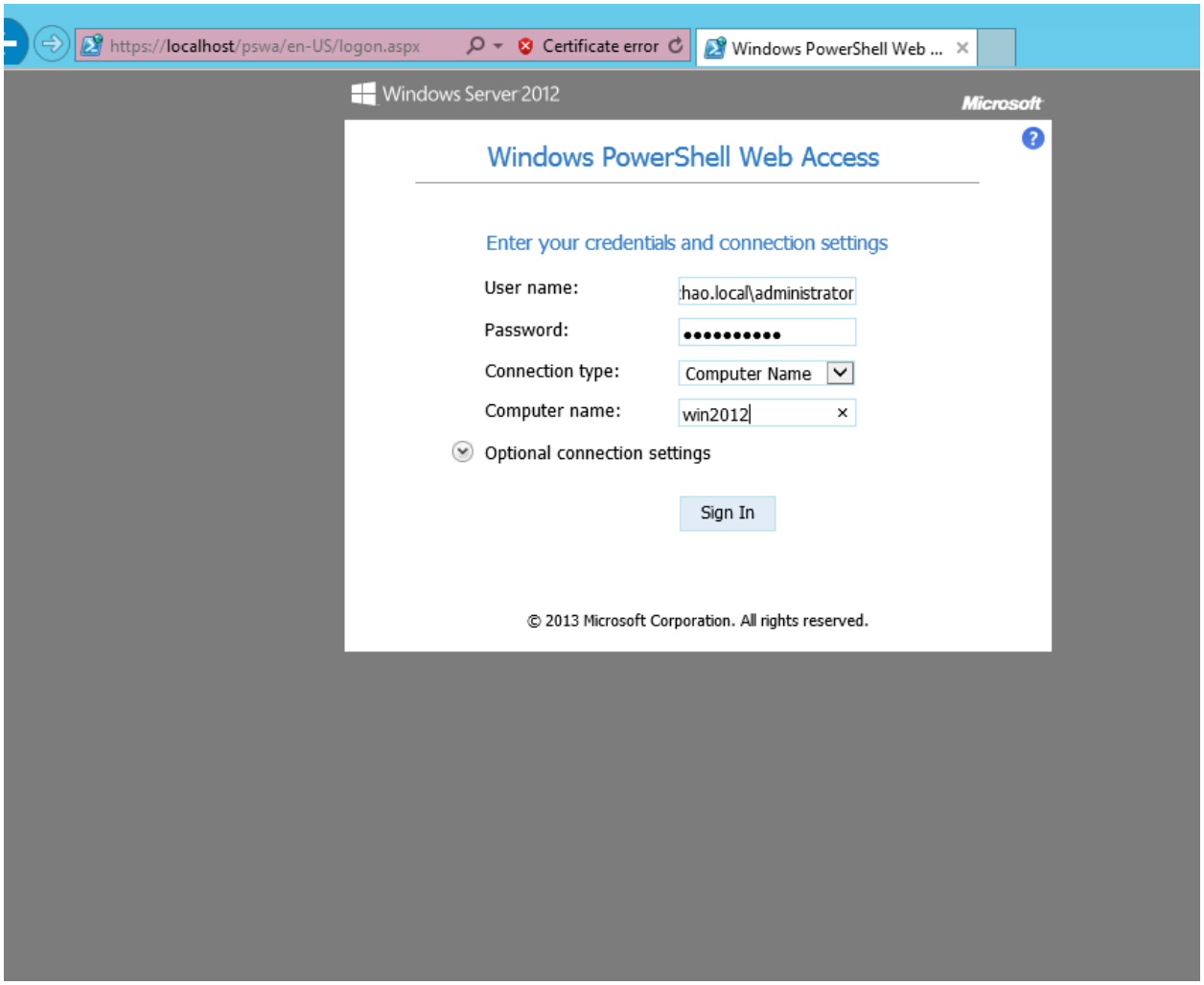
$authorizationFile = "C:\windows\web\powershellwebaccess\data\AuthorizationRules.xml"

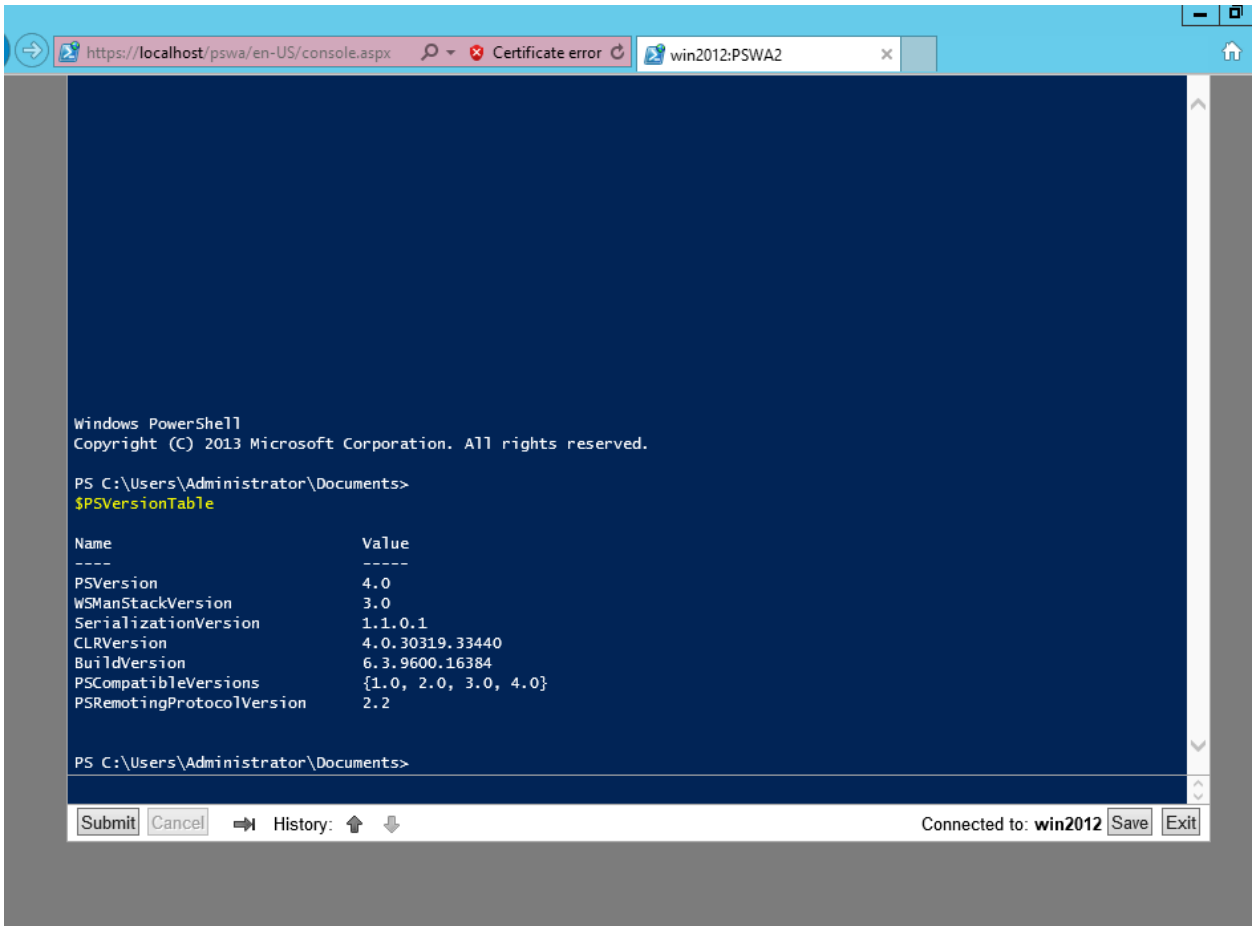
c:\windows\system32\icacls.exe $authorizationFile /grant ('"' + "IIS AppPool\$applicationPoolName"

+ '":R') > $null



4. Testing results (bonus)





5. Summary

Need ADDC to implement authentication process.

And *ConfigurationName* has to be *Microsoft.PowerShell*

6. Reference

http://vcloud-lab.com/entries/powershell/setup-and-configure-powershell-web-access-server-gateway-

https://www.jorgebernhardt.com/how-to-install-windows-powershell-web-access-gateway/